

CMSC427

Transformations:

Matrix Review

Credit: slides 9+ from Prof. Zwicker

Matrix practice

$$M = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \quad MR = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix} =$$

$$R = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix} \quad RM = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} =$$

$$P = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad MP = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} =$$

Matrix practice

$$M = \begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix} \quad MR = \begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix} = \begin{bmatrix} 2 & 7 \\ 1 & 6 \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix} \quad RM = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} 3 & 2 \\ 3 & 6 \end{bmatrix}$$

$$P = \begin{bmatrix} 2 \\ 3 \end{bmatrix} \quad MP = \begin{bmatrix} 2 & 0 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

Matrix transpose and column vectors

$$R^T = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix}^T =$$

$$H^T = \begin{bmatrix} 2 & 1 & 3 \\ 4 & 1 & 5 \end{bmatrix}^T =$$

$$P = \begin{bmatrix} 2 \\ 3 \end{bmatrix} = [2 \quad 3]^T$$

Matrix transpose and column vectors

$$R^T = \begin{bmatrix} 1 & 1 \\ 0 & 3 \end{bmatrix}^T = \begin{bmatrix} 1 & 0 \\ 1 & 3 \end{bmatrix}$$

$$H^T = \begin{bmatrix} 2 & 1 & 3 \\ 4 & 1 & 5 \end{bmatrix}^T = \begin{bmatrix} 2 & 4 \\ 1 & 1 \\ 3 & 5 \end{bmatrix}$$

$$P = \begin{bmatrix} 2 \\ 3 \end{bmatrix} = [2 \quad 3]^T$$

Abstract point of view

- Mathematical objects with set of operations
 - Addition, subtraction, multiplication, multiplicative inverse, etc.
- Similar to integers, real numbers, etc.

But

- Properties of operations are different
 - E.g., multiplication is not commutative
- Represent different intuitive concepts
 - Scalar **numbers** represent **distances**
 - **Matrices** can represent **coordinate systems, rigid motions**, in 3D and higher dimensions, etc.

Practical point of view

- Rectangular array of numbers

$$\mathbf{M} = \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & m_{1,n} \\ m_{2,1} & m_{2,2} & \dots & m_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{m,1} & m_{2,2} & \dots & m_{m,n} \end{bmatrix} \in \mathbf{R}^{m \times n}$$

- Square matrix if $\mathbf{m} = \mathbf{n}$
- In graphics often $\mathbf{m} = \mathbf{n} = 3, \mathbf{m} = \mathbf{n} = 4$

Matrix addition

$$\mathbf{A} + \mathbf{B} = \begin{bmatrix} a_{1,1} + b_{1,1} & a_{1,2} + b_{1,2} & \dots & a_{1,n} + b_{1,n} \\ a_{2,1} + b_{2,1} & a_{2,2} + b_{2,2} & \dots & a_{2,n} + b_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} + b_{m,1} & a_{2,2} + b_{2,2} & \dots & a_{m,n} + b_{m,n} \end{bmatrix}$$

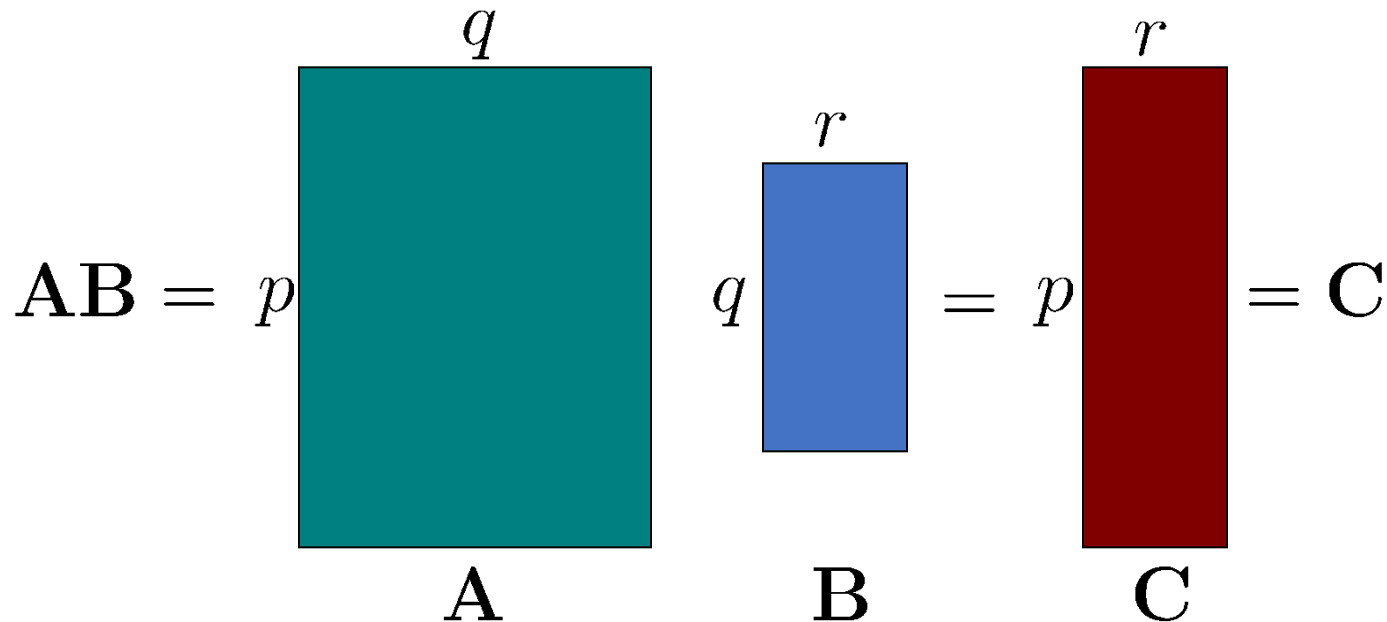
$$\mathbf{A}, \mathbf{B} \in \mathbf{R}^{m \times n}$$

Multiplication with scalar

$${}_s\mathbf{M} = \mathbf{M}_s = \begin{bmatrix} sm_{1,1} & sm_{1,2} & \dots & sm_{1,n} \\ sm_{2,1} & sm_{2,2} & \dots & sm_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ sm_{m,1} & sm_{2,2} & \dots & sm_{m,n} \end{bmatrix}$$

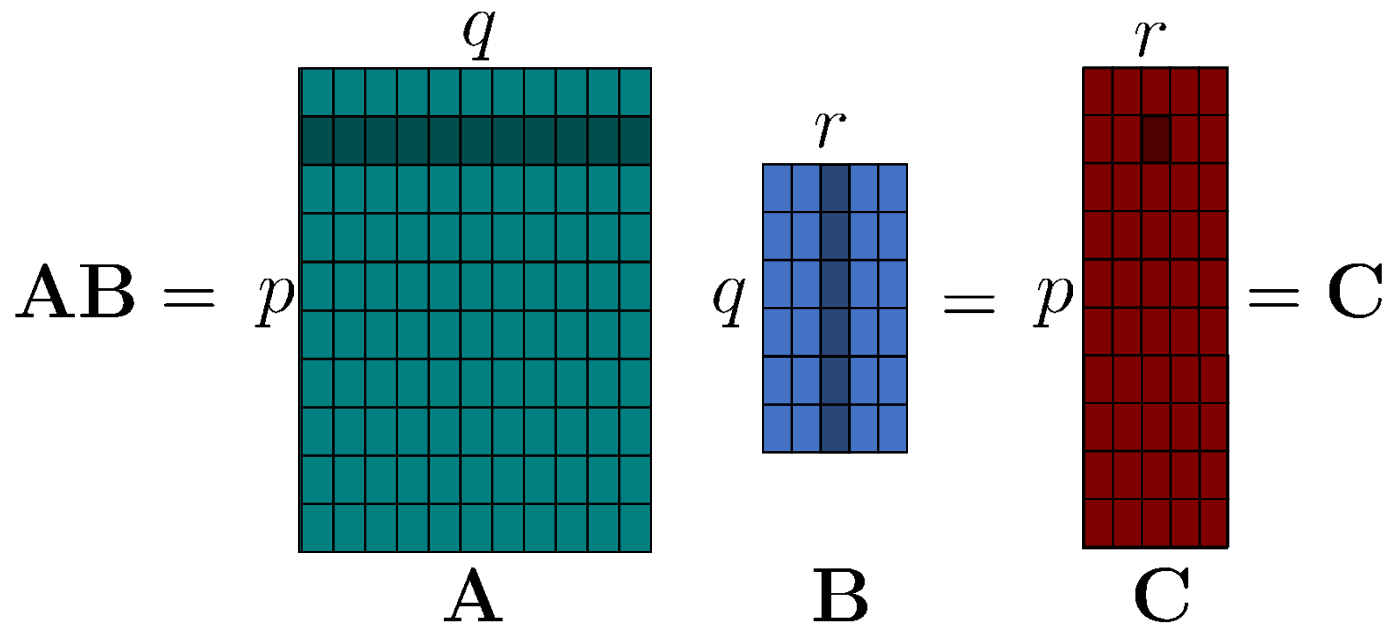
Matrix multiplication

$$\mathbf{AB} = \mathbf{C}, \quad \mathbf{A} \in \mathbf{R}^{p,q}, \mathbf{B} \in \mathbf{R}^{q,r}, \mathbf{C} \in \mathbf{R}^{p,r}$$



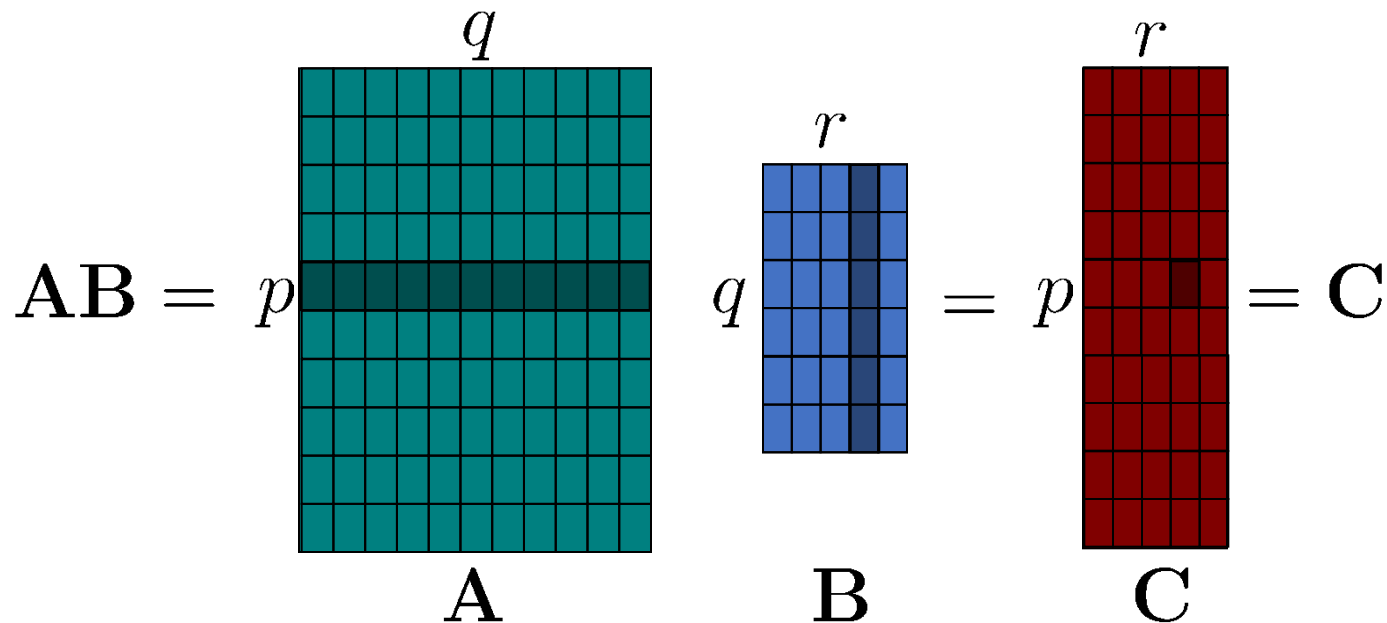
Matrix multiplication

$$\mathbf{AB} = \mathbf{C}, \quad \mathbf{A} \in \mathbf{R}^{p,q}, \mathbf{B} \in \mathbf{R}^{q,r}, \mathbf{C} \in \mathbf{R}^{p,r}$$



Matrix multiplication

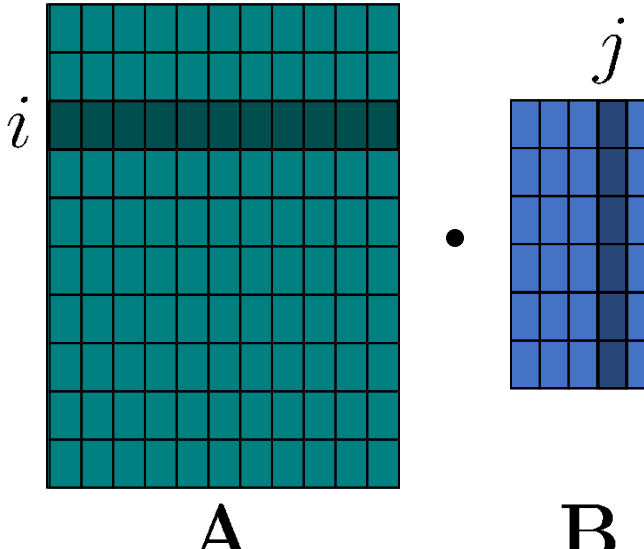
$$\mathbf{AB} = \mathbf{C}, \quad \mathbf{A} \in \mathbf{R}^{p,q}, \mathbf{B} \in \mathbf{R}^{q,r}, \mathbf{C} \in \mathbf{R}^{p,r}$$



Matrix multiplication

$$\mathbf{AB} = \mathbf{C}, \quad \mathbf{A} \in \mathbf{R}^{p,q}, \mathbf{B} \in \mathbf{R}^{q,r}, \mathbf{C} \in \mathbf{R}^{p,r}$$

$$(\mathbf{AB})_{i,j} = \mathbf{C}_{i,j} = \sum_{k=1}^q a_{i,k} b_{k,j}, \quad i \in 1..p, j \in 1..r$$

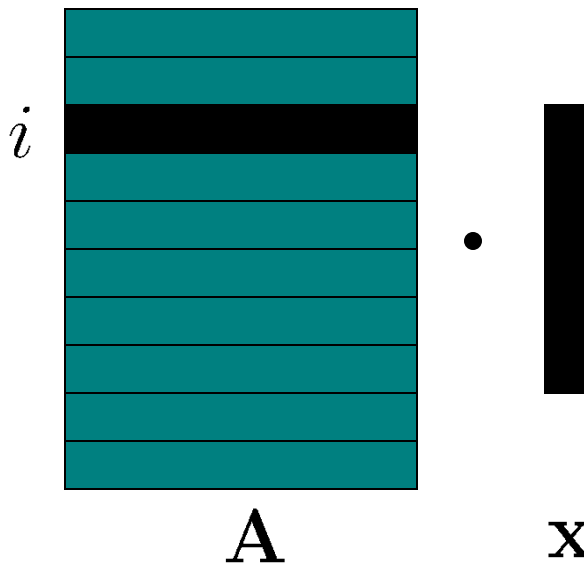
$$(\mathbf{AB})_{i,j} = \mathbf{C}_{i,j} =$$


The diagram shows a teal grid representing matrix A with 10 rows and 10 columns. The 5th row is highlighted in a darker teal and labeled with the index i to its left. Below the grid is the label **A**. To the right of matrix A is a blue grid representing matrix B with 10 rows and 5 columns. The 5th column is highlighted in a darker blue and labeled with the index j above it. Below the grid is the label **B**. A black dot \cdot is placed between the two grids, indicating the dot product operation.

Special case: matrix-vector multiplication

$$\mathbf{Ax} = \mathbf{y}, \quad \mathbf{A} \in \mathbf{R}^{p,q}, \mathbf{x} \in \mathbf{R}^q, \mathbf{y} \in \mathbf{R}^p$$

$$(\mathbf{Ax})_i = \mathbf{y}_i = \sum_{k=1}^q a_{i,k} x_k$$

$$(\mathbf{Ax})_i = \mathbf{y}_i =$$


The diagram shows a vertical stack of teal horizontal bars representing the rows of matrix \mathbf{A} . The i -th bar from the top is highlighted in black. To the right of this bar is a black dot, and further right is a single black vertical bar representing the vector \mathbf{x} . Below the stack of bars is the label \mathbf{A} , and below the single bar is the label \mathbf{x} .

- Distributive law holds

i.e., matrix $\mathbf{A}(s\mathbf{B} + t\mathbf{C}) = s\mathbf{AB} + t\mathbf{AC}$

http://en.wikipedia.org/wiki/Linear_map

- But multiplication is **not commutative**,

in general

$$\mathbf{AB} \neq \mathbf{BA}$$

Identity matrix

$$\mathbf{I} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \in \mathbf{R}^{n \times n}$$

$$\mathbf{MI} = \mathbf{IM} = \mathbf{M}, \quad \text{for any } \mathbf{M} \in \mathbf{R}^{n \times n}$$

Definition

If a square matrix \mathbf{M} is non-singular, there exists a unique **inverse** \mathbf{M}^{-1} such that

$$\mathbf{M}\mathbf{M}^{-1} = \mathbf{M}^{-1}\mathbf{M} = \mathbf{I}$$

- Note

$$(\mathbf{M}\mathbf{P}\mathbf{Q})^{-1} = \mathbf{Q}^{-1}\mathbf{P}^{-1}\mathbf{M}^{-1}$$

- Computation

- Gaussian elimination, Cramer's rule (OctaveOnline)
- Review in your linear algebra book, or quick summary <http://www.maths.surrey.ac.uk/explore/emmaspages/option1.html>

Java vs. OpenGL matrices

- OpenGL (underlying 3D graphics API used in the Java code, more later)

<http://en.wikipedia.org/wiki/OpenGL>

- Matrix elements stored in array of floats `float M[16];`
 - “Column major” ordering
- Java base code
 - “Row major” indexing
 - Conversion from Java to OpenGL convention hidden somewhere in basecode!

$$\begin{bmatrix} m[0] & m[4] & m[8] & m[12] \\ m[1] & m[5] & m[9] & m[13] \\ m[2] & m[6] & m[10] & m[14] \\ m[3] & m[7] & m[11] & m[15] \end{bmatrix}$$

$$\begin{bmatrix} m(0,0) & m(0,1) & m(0,2) & m(0,3) \\ m(1,0) & m(1,1) & m(1,2) & m(1,3) \\ m(2,0) & m(2,1) & m(2,2) & m(2,3) \\ m(3,0) & m(3,1) & m(3,2) & m(3,3) \end{bmatrix}$$